

METHOD AND APPARATUS FOR ENCODING DATA TO GUARANTEE
ISOLATED TRANSITIONS IN A MAGNETIC RECORDING SYSTEM

RELATED APPLICATIONS

- 5 [0001] This application is related to the copending and commonly assigned United States patent application entitled "Method, System, and Program for Synchronization and Resynchronization of a Data Stream", having attorney docket no. TUC920010036US1, which patent application was filed on the same date herewith and is incorporated herein by reference in its entirety.

10

BACKGROUND OF THE INVENTION

1. Field of the Invention

- [0002] The present invention relates to a method and apparatus for storing data in a storage medium. More specifically, the present invention relates to gain and timing control in storing data in a storage medium.
- 15

2. Description of the Related Art

- [0003] In data recording systems, a data drive writes positive and negative "flux transitions" to the medium. A "one" bit ("1") represents a peak or trough in the signal while a "zero" bit ("0") indicates that no signal is present. These flux transitions within a data stream can be used to provide feedback for timing and gain control loops. However, if a string of zeros exist for too long, a phase change may not be detected, causing errors in the recording system. This problem can be avoided if the data is encoded so that a "1" is guaranteed to occur at a definite minimum frequency. This is the purpose of modulation coding subject to a classical runlength limited k-constraint.
- 20
- 25

- [0004] But there may be advantages to using timing marks or gain control marks other than the symbol "1." In the 1999 publication "One-Pairs Codes for Partial Response Magnetic Recording," IEEE Transactions on Magnetism, Vol. 35, No. 3,

May 1999, the use of a pair of 1s (i.e. "11") is described to perform timing recovery for readback of information stored on magnetic recording media in a partial response channel. Still other channel models may benefit from a different encoding system to provide additional features, but the systems are limited to the available control marks and coding blocks in the prior art.

[0005] Thus, there is a need in the art to provide more sophisticated timing and gain control marks and/or improved coding algorithms for encoding and storing data in a storage medium.

10 SUMMARY OF THE PREFERRED EMBODIMENTS

[0006] Provided is a method, system, and program for storing input groups of uncoded binary data on a storage medium. A plurality of uncoded data blocks in a data stream are received. An encoded data stream is obtained from concatenating successive encoded blocks such that the encoded data stream includes a
15 predetermined bit pattern comprising a plurality of bits. The bit pattern always occurs within a first number of bits and two occurrences of a "1" or "0" occur within a second number of bits. The encoded data stream is stored on the storage medium.

[0007] In further implementations, the predetermined bit pattern comprises "010". In such case, each uncoded data block may comprise eight bits and each encoded data
20 block may comprise nine bits. Still further, each uncoded data block may comprise sixteen bits and each encoded data block may comprise seventeen bits.

[0008] In still further implementations, the predetermined bit pattern may comprise "111". In such case, each uncoded data block comprises nine bits and each encoded data block comprises ten bits.

25 [0009] Still further, the predetermined bit pattern may comprise either "0100" or "0010". In such case, each uncoded data block comprises sixteen bits, each encoded data block comprises seventeen bits, and the first number comprises fifteen bits.

[0010] In further implementations, the predetermined bit pattern comprises 111 and the m/n rate coded block comprises a 9/10 rate coded block. In still further

implementation, the predetermined bit pattern comprises either 0010 or 0100, and the m/n rate coded block comprises a 9/10 rate coded block.

[0011] The described implementations provide a technique to encode uncoded binary data at a guaranteed minimum frequency rate using predetermined binary
5 patterns representing peaks in an analog waveform which can provide improved timing and gain control.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] Referring now to the drawings in which like reference numbers represent
10 corresponding parts throughout:

FIG. 1 illustrates a storage environment in which aspects of the invention are implemented;

FIG. 2 illustrates a data flow implemented in the encoder to encode and store
15 a block of uncoded binary user data in accordance with implementations of the invention;

FIG. 3 illustrates an encoding table using a "010" binary pattern in accordance with implementations of the invention;

FIG. 4 illustrates the maximum separation between consecutive peaks embodied by the "010" binary pattern in accordance with implementations of the
20 invention;

FIG. 5 illustrates an encoding table using a "111" binary pattern in accordance with implementations of the invention; and

FIG. 6 illustrates an encoding table using a "0100" or "0010" binary patterns in accordance with implementations of the invention.

25

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0013] In the following description, reference is made to the accompanying drawings which form a part hereof and which illustrate several embodiments of the present invention. It is understood that other embodiments may be utilized and

structural and operational changes may be made without departing from the scope of the present invention.

[0014] FIG. 1 illustrates an tape storage environment in which aspects of the invention are implemented. A host system 2 is in communication with a tape drive 4. The tape drive 4 may be a component within the host system 2 enclosure or a drive within a tape library or tape server that the host system 2 communicates with over a network (not shown). The tape drive 4 includes an encoder 6 to encode data received from the host 2 that is to be written on a tape medium 8 in a tape cartridge 10 engaged with the tape drive 4. The tape drive 4 further includes a decoder 12 to decode data stored on the tape medium 8 to return to the host system 2. A controller 14 within the tape drive 4 drives a read/write mechanism 16 to perform read and write operations with respect to encoded data on the tape medium 8 in a manner known in the art. The encoder 6 and decoder 12 may be implemented as separate hardware components external to the controller 14 or implemented within logic executed by the controller 14.

[0015] In alternative implementations, the tape drive 4 may comprise other types of storage devices, such as a hard disk drive, optical disk or other device for reading and writing data to a non-volatile storage medium. In the described implementations, the tape medium 8 comprises a magnetic or digital tape.

[0016] FIG. 2 illustrates the data flow implemented in the encoder 6 to encode and store a block of uncoded binary user data in accordance with one implementation of the invention. Control begins at block 100 when the encoder 6 receives a block of binary user data for storage on the medium 100. Next, at block 102, the encoder 104 encodes the block of binary user data in preparation for storage on the tape medium 8. Specifically, the encoder 104 encodes each word of the binary data block using an “m/n rate code block.” Under this coding scheme, which is shown in FIG. 3, “m” represents the number of bits in a “group” of uncoded binary user bits 310 to be encoded, and the “n” represents the number of bits in the corresponding “group” of encoded bits 320. Each group of “n” encoded bits 320 contains at least one binary

pattern that enables improved reliability of gain and timing control operations, as discussed below in more detail. After the encoder 6 individually encodes groups of uncoded binary user data 310, the encoded data 320 is stored on the tape medium 8 at block 104. More particularly, the read/write mechanism 16 stores binary flux

5 transitions corresponding to the encoded binary bit block to the recording medium 100. The storing of the data can be performed using known techniques in the art. [0017] FIG. 3 illustrates a modulation code table 300 implemented in the encoder 6 and decoder 12 to respectively encode and decode data. The modulation code table 300 provides a high rate modulation to encode the arbitrary binary data blocks 310
10 into encoded blocks 320 at an 8/9 code rate (i.e. a "group" of uncoded bits includes 8 bits and a "group" of encoded bits includes 9 bits). More specifically, a "group" of uncoded binary user data occupies 8 bits, encompassing all possible input combinations from "00000000" to "11111111." Each "group" of encoded bits occupies nine bits, and contains at least one predetermined binary pattern, also
15 referred to as "timing (or gain) control marks" or simply "marks." In the implementation of FIG. 3, the predetermined binary pattern "010" is used, which represents an isolated peak in an analog readback waveform. For extended partial response channels, the pattern "010" provides much greater reliability than simply a "1" in the classical k-constraint or "11" described in the IEEE publication "One-Pairs
20 Codes for Partial Response Magnetic Recording."

[0018] In the implementation of FIG. 3, the predetermined pattern "010" is guaranteed to occur within each encoded data block 320 such that the binary patterns (i.e. "010") from two neighboring encoded groups cannot be separated by more than 12 intervening bits in a traditional PR4 ("Partial Response") system, as shown by
25 FIG. 4. Further, with the modulation pattern of FIG. 3, the maximum gap between two occurrences of "1" and two occurrences of "0" is 6 bits, as can be shown by examining FIG. 3. Accordingly, the "010" pattern occurs in the encoded bit stream with a guaranteed number of bits and any two occurrences of "1" and two occurrences of "0" occur within a number of bits less than the maximum

number of bits between the "010" pattern. This ensures that a maximum possible amplitude occurs between instances of the predetermined number of bits, i.e., predetermined frequency, in a manner that isolates peaks and aid in analog gain control and digital timing recovery. Also, because the maximum gap between two

5 occurrences of "1" and two occurrences of "0" is 6 bits, this code satisfies a traditional PRML G constraint, which aids in traditional PR4 timing algorithms. Moreover, this code satisfies a traditional I constraint, which helps to limit the Viterbi path memory. Moreover, the code avoids an indefinite run of the VFO field (by excluding the two words (010101010 and 101010101).

10 **[0019]** In designing a code, the following goals should be considered: the ease with which the code can be decoded; the compatibility of the binary pattern with the system; occurrence at a sufficiently a high frequency, or within a relatively low predetermined number of bits; and the ability to consider noise enhancement, channel impulse response, and implementation complexity. The code described in
15 FIG. 3 provides a high frequency rate, low complexity, and very little error propagation.

[0020] The 8/9 block codes described in FIG. 3 can readily be extended to an extended block code having a 16/17 bit code rate. By appending additional eight bits to the encoded nine bits by alternating encoded bytes with uncoded bytes, a block
20 code using a 16/17 bit code rate is also available. With this 16/17 bit encoding scheme, the binary patterns (i.e. "010") from two neighboring encoded groups cannot be separated by more than 20 intervening bits in a traditional PR4 ("Partial Response") system, and the maximum gap between two occurrences of "1" and two occurrences of "0" is 14 bits.

25 **[0021]** The 16/17 bit code rate may be implemented as a block coded sequence or finite state code sequence. In the block coded sequence implementation, there is a one-to-one correspondence of uncoded blocks to encoded blocks. In a finite state code implementation, the same uncoded block may be represented by two different

encoded blocks or one encoded block can represent two different uncoded blocks, and ambiguity is resolved by looking at adjacent blocks.

[0022] FIG. 5 illustrates an encoding table using an alternative "111" binary pattern in accordance with implementations of the invention. In the preferred

5 implementations, a high rate modulation code 500 is used to encode the arbitrary binary data blocks 510 into encoded blocks 520 at a 9/10 code rate (i.e. a "group" of uncoded bits includes 9 bits and a "group" of encoded bits comprises 10 bits). Each subscripted "a" character refers to the value of a bit of the uncoded group 510, where the subscript identifies the bit's position in the group 510. For example, a_0 identifies

10 the binary "0" or "1" located at the first bit position of the uncoded group 510. In FIG. 5, all possible uncoded groups 510 and the corresponding output encoded groups 520 are shown. In the PR4 system, the first and last ones must have opposite signs, on either side of a waveform peak, and two consecutive ones of opposite sign, on either side of a zero crossing. For instance, if the first one in "111" corresponds to

15 +1, then the third one corresponds to -1. Thus, there are going to be two consecutive ones having the same sign, corresponding to a peak in PR4. The modulation code 500 ensures that the binary pattern "111" appears in each encoded data block 520. With this encoding scheme, the binary patterns (i.e. "111") from two neighboring encoded groups cannot be separated by more than 14 intervening bits at a 9/10 bit

20 code rate and 21 intervening bits in an extended 16/17 bit code rate. In order to obtain a 16/17 code, seven bits are added. In certain implementations, the encoding table of FIG. 5 is to be used for all nine bit uncoded blocks except for the block "001111111", which encodes instead to the ten bit block "0110000111". This ensures that every codeword contains both a "0" and a "1" Otherwise, other factors

25 must be used to select the uncoded to encoded block correspondence, such as the case with a finite state coding.

[0023] FIG. 6 illustrates an encoding table using an alternative "0100" or "0010" binary patterns in accordance with further implementations of the invention. In the preferred implementations, a high rate modulation code 600 is used to encode the

arbitrary binary data blocks 610 into encoded blocks 620 at a 9/10 code rate (i.e. a “group” of uncoded bits includes 9 bits and a “group” of encoded bits includes 10 bits). As in FIG. 5, each subscripted “a” character refers to the value of a bit of the uncoded group 610, where the subscript identifies the bit’s position in the group 610.

- 5 In FIG. 6, all possible uncoded groups 610 and the corresponding output encoded groups 620 are shown. The modulation code 600 ensures that the binary pattern “0100” or “0010” appears in each encoded data block 520. The binary patterns “0100” or “0010” are suitable for implementation in NRZI, giving peaks both in EPR4 (“Extended Partial Response”) and E²PR4 (“Extended Partial Response 2”) systems. Moreover, using longer binary patterns “0100” or “0010” can increase the frequency of the timing mark in the encoded data. By using alternative timing marks within a single block code, the binary patterns (i.e. “0100” or “0010”) from two neighboring encoded groups cannot be separated by more than 12 intervening bits for the 9/10 bit code rate and 19 intervening bits for the extended 16/17 bit code rate, rather than 20 intervening bits in a code block using a single timing mark.
- 10
15

Additional Implementation Details

- [0024] The preferred embodiments may be implemented as a method, apparatus or article of manufacture using standard programming and/or engineering techniques to produce software, firmware, hardware, or any combination thereof. The term “article of manufacture” as used herein refers to code or logic implemented in hardware logic (e.g., an integrated circuit chip, Field Programmable Gate Array (FPGA), Application Specific Integrated Circuit (ASIC), etc.) or a computer readable medium (e.g., magnetic storage medium (e.g., hard disk drives, floppy disks, tape, etc.), optical storage (CD-ROMs, optical disks, etc.), volatile and non-volatile memory devices (e.g., EEPROMs, ROMs, PROMs, RAMs, DRAMs, SRAMs, firmware, programmable logic, etc.). Code in the computer readable medium is accessed and executed by a processor. The code in which preferred embodiments are implemented may further be accessible through a transmission media or from a file server over a
- 20
25

network. In such cases, the article of manufacture in which the code is implemented may comprise a transmission media, such as a network transmission line, wireless transmission media, signals propagating through space, radio waves, infrared signals, etc. Of course, those skilled in the art will recognize that many modifications may be made to this configuration without departing from the scope of the present invention, and that the article of manufacture may comprise any information bearing medium known in the art.

[0025] In the described implementations, the encoding process was described with respect to encoding the uncoded binary user data into encoded data. The present invention also encompasses decoding the encoded data back to the uncoded binary user data using the same block codes because the encoders are one-to-one correspondences. Additional hardware may be used in the readback process including a decoder. More specifically, the decoder decodes the encoded bit stream by reversing the translation of FIG. 3 to effectively decode each encoded group of bits back into an uncoded 8-bit group. Similarly, the decoder can reverse the translation of FIG. 5 and 6 to effectively decode each encoded group of bits back into an uncoded 9-bit group. For finite-state codes, certain encoded words may represent more than one possible uncoded bit group, the decoder may apply known methods (i.e. look at the next code word) to determine the correct translation.

[0026] In certain described implementations, the encoder tables provide a one-to-one correspondence of uncoded to encoded blocks. In alternative implementations, finite-state codes can be used instead of the block code using finite-state encoders. A finite-state encoder will encode each user data block into a block that satisfies the given constraint of the system at some rate m/n . Each m -bit user input is encoded into an n -bit codeword as a function of the current state (as well as the user input), wherein the state transition consists of an initial state, terminal state, m -bit input and n -bit codeword. In finite-state coding schemes, the same encoded codeword can correspond to two different uncoded user data blocks (providing such benefits as a higher frequency rate and smaller gap distance between timing marks vs. the costs of

increased complexity) and one uncoded block can correspond to two encoded blocks. In finite-state codes, the state information is used to determine how to properly decode the encoded data, i.e., by using the value of adjacent blocks to determine the uncoded to coded block mapping.

- 5 [0027] The described implementations provide a technique for transferring data to a tape drive. The above described logic may be used with other input/output (I/O) devices or other storage devices, e.g., optical tape, magnetic tape, magnetic disk, etc.

- [0028] The logic implementation of FIG. 2 described specific operations as occurring in a particular order. In alternative implementations, certain of the flow
10 operations may be performed in a different order, modified or removed and still implement preferred embodiments of the present invention. Moreover, steps may be added to the above described flow and still conform to implementations of the invention.

- [0029] The foregoing description of the preferred embodiments of the invention has
15 been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. It is intended that the scope of the invention be limited not by this detailed description, but rather by the claims appended hereto. The above specification, examples and data provide a
20 complete description of the manufacture and use of the composition of the invention. Since many embodiments of the invention can be made without departing from the spirit and scope of the invention, the invention resides in the claims hereinafter appended.